Monte Hall problem

> *Suppose you're on a game show, and you're given the choice of three doors: Behind one door is a car; behind the others, goats. You pick a door, say No. 1, and the host, who knows what's behind the doors, opens another door, say No. 3, which has a goat. He then says to you, "Do you want to pick door No. 2?" Is it to your advantage to switch your choice?*
>
> (From https://en.wikipedia.org/wiki/Monty_Hall_problem)

Description

> Write a program that runs a simulation demonstrating the solution to the Monte Hall problem, which is that as a contestant it is always to your advantage to switch your original door choice. Demonstrate that contestants who switch have a 2/3 chance of winning the car.

Requirements

- Define a function named `simulate()` that runs a complete simulation, including at least 100,00 trials. This function should declare at least two variables named `nstay` and `nswitch`.
- Define a function named `play()` within `simulate()`. This function will run one trial of the Monte Hall problem and increment either `nstay`, if it was to the contestant's advantage to stay with her original door choice, or `nswitch`, if it was to the contestant's advantage to switch her original choice.
- The `play()` function should have these steps:
    1. Generate a `doors` Array with randomly ordered `"goat"`, `"goat"`, `"car"` strings.
    2. Randomly choose one of the three doors and note door number a contestant's original selection.
    3. Find a `"goat"` in the remaining two unselected doors. This door is now effectively opened and can no longer be selected.
    4. Find the index of the remaining unselected-unopened door.
    5. If the `"car"` is behind the originally selected door, it is to the contestant's advantage to stay with her original door selection. Increment the `nstay` variable.
    6. If the `"car"` is behind the unselected-unopened door, it is to the contestant's advantage to switch her original door selection. Increment the `nswitch` variable.
- Run at least 100,000 iterations of `play()`.
- Compute and print final fractions by dividing `nstay` and `nswitch` each by the total number of iterations performed.
- Kick off the simulation by invoking `simulate()`.

Hints

- Consider writing a `randomInt(min, max)` function that returns a random integer between `min` and `max`.

Finishing Up

- You MUST enter header comments into you JavaScript file including (1) File name, (2) Your name, (3) Description and or purpose of the assignment
- You MUST comment you code, explaining what you did in each section
- Submit your single JavaScript file using Canvas under the appropriate assignment name